X3T9.2/93-110 Rev

To: _sff
cc:
From: Hale Landis
Date: 06/21/93 04:50:43 PM
Subject: Comments on ATA Rev 4 DMA Definitions

Date: 21 June 93

From: Hale Landis, Seagate
Ph 498-439-2443 Fx 408-438-4190
Email hale_landis@notes.seagate.com

To: SFF Membership (via SFF Reflector)

Subject: Comments on ATA Rev 4 DMA Definitions

There appear to be a number of problems with the current definition of DMA. Seagate would like to bring these to the attention of the ATA and/or SFF committees for discussion and resolution. It is unclear to us where the proper forum is to discuss these problems. ATA Rev 4 is going out for another public review and perhaps that is when we should comment on this issue? Or is this something that belongs in the ATA Extension committee? However, the ATA/SFF Local Bus WG seems to be addressing some DMA issues at this time (even though the original charter of this WG was to address only "high speed PIO" issues), maybe the ATA/SFF LB WG would like to address this too? Perhaps you can understand our confusion.

Here are the problems we have found to date listed by ATA section number:

* 5.4 I/O Cable -- In Table 3, Cable Parameters, the IoH current is specified as 400uA. Is this adequate to meet the timing requirements of DMARQ (and is it adequate to meet the timing requirements of the other signals) assuming a 200pF load?

* 6.3.9 DMARQ -- We believe this signal requires a tristate definition in order to support DMA on master and slave. What is the value and the location of the required pull-down resistor? EISA specifies a 5.6K resistor on the motherboard.

* 7.2.6 Device Control Register -- If DMARQ requires tristate then bit 0 of this register could be redefined to control the driver for DMARQ. We understand that some controllers used bit 0 to disable (or tristate) the DMARQ driver (bit 0 = 0 enables, bit 0 = 1 to disable). Something like this is required if the ATA controller must share a DMA channel with another controller.

* 7.2.13 Status Register -- Either the Busy or the DRQ bit must be 1 during the DMA Data Phase. See comment about section 10.5 below.

* 9.9.12 Word 52: DMA data transfer cycle timing mode -- If words 62 or 63 are used, what value should be placed into word 52?

* 9.15 Read DMA -- This section states that on an unrecoverable error, the sector in error is not transferred. We think Read DMA should be like Read Multiple in which the sector with the unrecoverable error may be transferred. We think the recommended host retry sequence should be similar to that recommend for Read Multiple. There is no description of what the Command Block register contents should be at the end of a Read DMA command.

* 9.22 Set Features --
  - What is "Block Mode"? There is no definition of this term in the ATA.
  - The relationship between Set Feature 03H and the DMA

Read/Write commands is unclear. Are DMA commands disabled until a Set Feature 03H command is executed?

* 9.28 Write DMA -- There is no description of what the Command Block register contents should be at the end of a Write DMA command.

* 10.5 DMA Data Transfer Command -- We assume that the DMA Data Phase is defined by the time during which the drive has DMARQ asserted. If this is correct, the statement "the register contents are not valid during a DMA Data Phase" is incorrect. The Status register MUST be valid and should have either the BUSY bit set to 1 or the DRQ bit set to 1 during the DMA Data Phase. There is nothing to prevent a host from reading the Status register during a time when DMARQ is asserted and DMACK is not asserted.

* 10.5.x Normal DMA Transfer, Aborted DMA Transfer and Aborted DMA Command -- These diagrams show confusing values for BSY and DRQ. Also these diagrams should use "Assert INTRQ" or "Negate INTRQ" instead of "nIEN=1" or "nIEN=0. We are not sure what nIEN=1 or nIEN=0 mean since nIEN is a bit in the Device Control register that only the host can modify.